

Clique in 3-track interval graphs is APX-hard*

Minghui Jiang

Department of Computer Science, Utah State University, Logan, UT 84322, USA
mjiang@cc.usu.edu

March 2, 2013

Abstract

Butman, Hermelin, Lewenstein, and Rawitz proved that CLIQUE in t -interval graphs is NP-hard for $t \geq 3$. We strengthen this result to show that CLIQUE in 3-track interval graphs is APX-hard.

Keywords: multiple-interval graphs, computational complexity.

1 Introduction

We prove the following theorem:

Theorem 1. *CLIQUE in 3-track interval graphs is APX-hard.*

Preliminaries Given a set $X = \{x_1, \dots, x_n\}$ of n boolean variables and a set $C = \{c_1, \dots, c_m\}$ of m clauses, where each variable occurs at most (resp. exactly) p times in the clauses, and each clause is the conjunction (resp. disjunction) of exactly q literals, p -OCC-MAX-Eq-CSAT (resp. Ep -OCC-MAX-Eq-SAT) is the problem of finding an assignment for X that satisfies the maximum number of clauses in C .

Lemma 1. *12-OCC-MAX-E2-CSAT is APX-hard.*

Proof. It is known that E3-OCC-MAX-E2-SAT is APX-hard [1]. For each disjunctive clause $x_1 \vee x_2$, we can construct a set of 6 conjunctive clauses

$$x_1 \wedge y \quad x_1 \wedge \bar{y} \quad x_2 \wedge y \quad x_2 \wedge \bar{y} \quad x_1 \wedge \bar{x}_2 \quad \bar{x}_1 \wedge x_2$$

where y is an additional dummy variable. If both x_1 and x_2 are false, then none of the 6 clauses is satisfied. If either x_1 or x_2 is true, then exactly 2 of the 6 clauses are satisfied. Thus we have a gap-preserving L-reduction [5] from E3-OCC-MAX-E2-SAT to 12-OCC-MAX-E2-CSAT with $\alpha = 2$ and $\beta = 1/2$. \square

2 Proof of Theorem 1

We prove that CLIQUE in 3-track interval graphs is APX-hard by an L-reduction from 12-OCC-MAX-E2-CSAT. Given an instance (X, C) of 12-OCC-MAX-E2-CSAT, we construct a 3-track interval graph G as the intersection graph of a set of $24n + m$ 3-track intervals:

*Supported in part by NSF grant DBI-0743670.

- 12 copies of a 3-track interval for the positive literal x_i of each variable $x_i \in X$;
- 12 copies of a 3-track interval for the negative literal \bar{x}_i of each variable $x_i \in X$;
- 1 copy of a 3-track interval for each clause $c_k \in C$.

Each 3-track interval in our construction is the union of three open intervals, one interval on each track, of integer endpoints between $-(n+1)$ and $n+1$.

For each variable x_i , the 3-track interval for the positive literal x_i is the union of the following three intervals

$$\text{track 1: } (-i, i) \quad \text{track 2: } (i, n+1) \quad \text{track 3: } (i, i+1)$$

and the 3-track interval for the negative literal \bar{x}_i is the union of the following three intervals

$$\text{track 1: } (i, n+1) \quad \text{track 2: } (-i, i) \quad \text{track 3: } (-(i+1), -i)$$

Assume without loss of generality that no clause contains both the positive literal and the negative literal of the same variable. For each clause c_k , we construct one 3-track interval following one of four cases:

1. $c_k = x_i \wedge x_j, i \leq j$.

$$\text{track 1: } (-(n+1), -j) \quad \text{track 2: } (-(n+1), i) \quad \text{track 3: } \begin{cases} (i+1, j) & \text{if } j > i+1 \\ (-1, 1) & \text{if } j = i \text{ or } i+1 \end{cases}$$

2. $c_k = \bar{x}_i \wedge \bar{x}_j, i \leq j$.

$$\text{track 1: } (-(n+1), i) \quad \text{track 2: } (-(n+1), -j) \quad \text{track 3: } \begin{cases} (-j, -(i+1)) & \text{if } j > i+1 \\ (-1, 1) & \text{if } j = i \text{ or } i+1 \end{cases}$$

3. $c_k = x_i \wedge \bar{x}_j, i < j$.

$$\text{track 1: } (i, j) \quad \text{track 2: } (-(n+1), -j) \quad \text{track 3: } (-1, i)$$

4. $c_k = \bar{x}_i \wedge x_j, i < j$.

$$\text{track 1: } (-(n+1), -j) \quad \text{track 2: } (i, j) \quad \text{track 3: } (-i, 1)$$

This completes the construction. We give an example in Figure 1. The reduction clearly runs in polynomial time. We have the following lemma:

Lemma 2. *There is an assignment for X that satisfies at least z clauses in C if and only if G has a clique of size at least $w = 12n + z$.*

Proof. The following observations can be easily verified:

- For any two variables x_i and x_j , $i \neq j$, the 3-track intervals for the literals of x_i overlap with the 3-track intervals for the literals of x_j .
- For any two clauses c_k and c_l , $k \neq l$, the 3-track interval for c_k overlaps with the 3-track interval for c_l .
- For each variable x_i , the 3-track intervals for the positive literal of x_i are disjoint from the 3-track intervals for the negative literal of x_i .

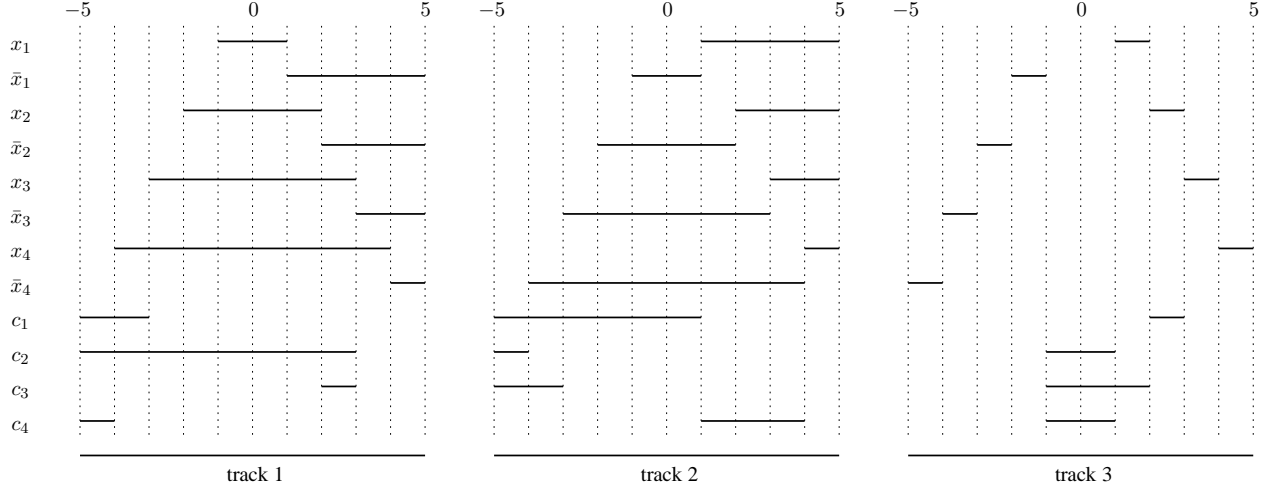


Figure 1: The set of 3-track intervals for a 12-OCC-MAX-E2-CSAT instance of $n = 4$ variables and $m = 4$ clauses $c_1 = x_1 \wedge x_3$, $c_2 = \bar{x}_3 \wedge \bar{x}_4$, $c_3 = x_2 \wedge \bar{x}_3$, and $c_4 = \bar{x}_1 \wedge x_4$. Duplicate 3-track intervals for each literal are omitted from the figure.

- For each clause c_k , the 3-track interval for c_k is disjoint from the 3-track intervals for the two literals in c_k , and overlaps with the 3-track intervals for the other literals.

We first prove the direction implication. Suppose there is an assignment for X that satisfies at least z clauses in C . We select a subset of pairwise-intersecting 3-track intervals as follows. For each clause c_k in C , select the corresponding 3-track interval if the clause is satisfied. Then, for each variable x_i in X , select the 12 copies of the 3-track interval for the negative literal of x_i if the variable is true, and select the 12 copies 3-track interval for the positive literal of x_i if the variable is false. Thus we obtain a clique of size at least $w = 12n + z$ in G .

We next prove the reverse implication. Suppose G has a clique of size at least $w = 12n + z$. Note that in our construction the number of 3-track intervals for each literal is at least the number of 3-track intervals for all clauses that contain the literal. Thus, by replacing vertices, any clique can be converted into a clique of at least the same size in *canonical* form, which includes, for each variable x_i in X , either all 12 copies of the 3-track interval for the positive literal of x_i , or all 12 copies of the 3-track interval for the negative literal of x_i . Assign x_i false if the clique includes the 3-track intervals for its positive literal, and assign x_i true if the clique includes the 3-track intervals for its negative literal. Thus we obtain an assignment for X that satisfies at least z clauses in C . \square

Let z^* be the maximum number of clauses in C that can be satisfied by an assignment of X , and let w^* be the maximum size of a clique in G . By Lemma 2, we have $w^* = 12n + z^*$. Since each clause in C is the conjunction of exactly two literals of the variables in X , we have $n \leq 2m$. Moreover, since a random assignment for X satisfies each clause in C with probability at least $1/4$, we have $z^* \geq m/4 \geq n/8$. It follows that

$$w^* = 12n + z^* \leq (8 \cdot 12 + 1)z^* = 97z^*.$$

Consider any clique of size w in G . Following the reverse implication in the proof of Lemma 2, we can find an assignment for X that satisfies at least $z = w - 12n$ clauses in C . Note that

$$|z^* - z| \leq |w^* - w|.$$

Thus we have an L-reduction from 12-OCC-MAX-E2-CSAT to CLIQUE in 3-track interval graphs with $\alpha = 97$ and $\beta = 1$. This completes the proof of Theorem 1.

Postscript This note was written in April 2010. The author would like to thank Stéphane Vialette for bringing the open questions of Butman et al. [2] to his attention in 2010, and for notifying him of the recent results of Francis et al. [4], who proved the APX-hardness of CLIQUE in several classes of multiple-interval graphs (including 3-track interval graphs) using the subdivision technique of [3].

References

- [1] P. Berman and M. Karpinski. Improved approximation lower bounds on small occurrence optimization. *Electronic Colloquium on Computational Complexity*, report no. 8, 2003.
- [2] A. Butman, D. Hermelin, M. Lewenstein, and D. Rawitz. Optimization problems in multiple-interval graphs. *ACM Transactions on Algorithms*, 6:#40, 2010. A preliminary version in *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'07)*, pages 268–277, 2007.
- [3] M. Chlebík and J. Chlebíková. The complexity of combinatorial optimization problems on d -dimensional boxes. *SIAM Journal on Discrete Mathematics*, 21:158–169, 2007.
- [4] M. C. Francis, D. Gonçalves, and P. Ochem. The maximum clique problem in multiple interval graphs. arXiv:1201.0043v2, 2012.
- [5] C. H. Papadimitriou and M. Yannakakis: Optimization, approximation, and complexity classes, *Journal of Computer and System Sciences*, **43** (1991), 425–440.